IN THE CLAIMS:

Please amend Claims 2, 3, 5 and 8.

Please add Claims 14-15 as follows.

1. (Amended) ~~A method for recovering 3D scene structure and camera motion from image data obtained from a multi-image sequence, wherein a reference image of the sequence is taken by a camera at a reference perspective and one or more successive images of the sequence are taken at one or more successive different perspectives by translating and/or rotating the camera, the method comprising the steps of:~~ An image processing method for recovery of a scene structure and camera motion from successive image data, comprising:

(a) comparing a ~~determining image data shifts for each successive image with respect to the~~ reference image to successive images, wherein the successive images are taken by translating or rotating the camera with respect to the reference image; ~~the shifts being derived from the camera translation and/or rotation from the reference perspective to the successive different perspectives;~~

(b) determining ~~constructing a shift data matrix that incorporates~~ the image data shifts for the successive ~~each~~ image with respect to the reference image;

(c) constructing a shift data representation that incorporates the image data shifts for each image including a first data record ~~calculating a rank-1 factorization from the shift data matrix using SVD, with one of the rank-1 factors being a vector~~ corresponding to the 3D structure and a second data record corresponding ~~the other rank-1 factor being a vector corresponding~~ to the ~~size of the~~ camera motions;

A4

(d) dividing the successive images into ~~smoothing~~ windows;

(e) determining ~~recovering~~ the direction of camera motion and 3D structure from the first and second data records between the windows. ~~vector corresponding to the 3D structure by solving a linear equation; and~~

~~(f) recovering the 3D structure by solving a linear equation using the recovered camera motion.~~

2. (Currently Amended) The method of claim 1, wherein, step (e) includes:

computing ~~a first~~ projection ~~matrix~~ matrices;

recovering camera rotation data records ~~vectors~~ and direction of camera translation from the shift data representation ~~matrix,~~ and ~~the first~~ projection matrices ~~matrix;~~

~~computing a second projection matrix; and~~

~~recovering the direction of camera translation using the shift data matrix, the reference image, the second projection matrix and the recovered camera rotation vectors~~.

3. (Currently Amended) The method of claim 2, wherein step ~~(f)~~ (e) further includes recovering the 3D structure from the shift data representation ~~matrix, the reference image~~, the recovered camera rotation data record ~~vectors~~ and the recovered direction of translation data record ~~vectors~~.

4. (Original) The method of claim 1, further including preliminary steps of:

recovering the rotations of the camera between each successive image; and

AH

warping all images in the sequence toward the reference image, while neglecting the translations.

5. (Currently Amended) The method of claim 1, wherein step (b) comprises:

computing H and Δ_{CH}, where H is a $(N_p - 3) \times N_p$ matrix defined so that HH^T is the identity matrix and H annihilates the three vectors Ψ_x, Ψ_y, Ψ_z where the three vectors are computed from the reference image as $\Psi_x \equiv \{\nabla I \cdot r^{(1)}(p)\}, \Psi_y \equiv \{\nabla I \cdot r^{(2)}(p)\}, \Psi_z \equiv \{\nabla I \cdot r^{(3)}(p)\}$ where

$r^{(1)}(x,y), r^{(2)}(x,y), r^{(3)}(x,y)$ are defined by $\left[r^{(1)}, r^{(2)}, r^{(3)}\right] \equiv \left[\begin{pmatrix} -xy \\ -(1+y^2) \end{pmatrix}, \begin{pmatrix} 1+x^2 \\ xy \end{pmatrix}, \begin{pmatrix} -y \\ x \end{pmatrix}\right]$

and Δ is a shift data matrix, that gives the difference in intensities between each successive image and the reference image and is a $(N_I - 1) \times N_p$ matrix with entries ΔI_n^i ,where ΔI_n^i is the change in (smoothed) intensity with respect to the reference image, and with no smoothing $\Delta I_n^i = I_n^i - I_n^0$, where N_I is the number of images, N_p is the number of pixels, and where I^i denotes the *i*-th image, with *i*=0.1..., $N_I - 1$, and where $I_n^i = I^i(p_n)$ denotes the image intensity at the *n*-th pixel position in I^i, where I^0 is the reference image, where *x* and *y* are the image coordinates of the pixel position and p= (*x*,*y*) and where $\Delta_{CH} \equiv C^{-1/2} \Delta H^T$ where *C* is a constant matrix ~~with~~ and where the notation {*V*} used to denote a vector with elements given by the V^a .

6. (Original) The method of claim 1, wherein step (c) comprises:

A4

computing a rank-1 factorization of $-\Delta_{CH} \approx \mathrm{M}^{(1)}\mathrm{S}^{(1)T}$ where $\mathrm{M}^{(1)}, \mathrm{S}^{(1)T}$ are vectors corresponding to the motion and structure respectively.

7. (Original) The method of claim 1, wherein step (c) comprises:

computing a rank-3 factorization of $-\Delta_{CH} \approx \sum_{a=1}^{3} \mathrm{M}^{(a)}\mathrm{S}^{(a)T}$ where $\mathrm{M}^{(a)}, \mathrm{S}^{(a)T}$ are vectors corresponding to the motion and structure respectively;

setting Z_n^{-1} as constant within each window, where Z is the depth from the camera to a 3D scene along the cameras optical axis;

listing each of the pixels so that those in the *k*-th smoothing window have sequential indices n_k, (n_k+1), . . . $(n_{k+1} - 1)$;

computing a first projection matrix by computing a Np x Np projection matrix P_Ω which is block diagonal with zero entries between different smoothing windows, and which annihilates the vectors $\{\nabla I \cdot \mathbf{p}\}, \{I_x\}$ and $\{I_y\}$ where $\{\nabla I\}$ is a vector containing the gradient of the intensity at each pixel, and I_x and I_y are the gradients of the intensity in the reference image in the *x* and *y* directions;

recovering the three camera rotation vectors includes solving the following equations

$\mathrm{P}_\Omega\left(\mathrm{H}^T\mathrm{S}^{(a)} - \Psi w^{(a)}\right) = 0$ for the 3-vector $\mathrm{w}^{(a)}$;

computing a second projection matrix includes computing a $N_p \times N_p$ projection matrix $\mathrm{P}_\mathrm{T}^{(a)}$, which is block diagonal with zero entries between different

smoothing windows and annihilates $\left(H^T S^{(a)}\right) - \Psi \mathrm{w}^{(a)}$ where $\mathrm{w}^{(a)}$ is the vector recovered previously;

recovering the directions of camera translations by solving for the directions of translation $\hat{\mathrm{T}}^{(a)}$ via

$$\mathrm{P}_{\hat{\mathrm{T}}}^{(a)}\left(-\hat{T}_x^{(a)}\{I_x\} - -\hat{T}_y^{(a)}\{I_y\} + \hat{T}_z^{(a)}\{\mathrm{p}\cdot\nabla I\}\right) = 0$$ and;

recovering Z_n via

$$\left(H^T S^{(a)}\right)_n - \left[\Psi w^{(a)}\right]_n = \tau^{(a)}\left(Z_n^{-1}\left(\hat{T}_z^{(a)}\mathrm{p}_n - \left\lfloor\hat{\mathrm{T}}^{(a)}\right\rfloor_2\right)\cdot\nabla I_n\right)$$ where $\left\lfloor\hat{\mathrm{T}}^{(a)}\right\rfloor_2$ represents $\hat{\mathrm{T}}_{\mathrm{x}}^{(a)}\hat{\mathrm{T}}_y^{(a)}$ the x and y component of the translation direction, and where $\tau^{(a)}$ are constants.

8. (Currently Amended) The method of claim 1, wherein step (d) comprises:

setting Z_n^{-1} as constant within each window, where Z is the depth from the camera to a 3D scene along the camera~~_~~'s optical axis;

listing each of the pixels so that those in the *k*-th smoothing window have sequential indices n_k, $(n_k+1), \ldots (n_{k+1} - 1)$.

9. (Original) The method of claim 2, wherein the step of computing a first projection matrix includes computing a Np x Np projection matrix P_{Ω} which is block diagonal with zero entries between different smoothing windows, and which annihilates the vectors $\{\nabla I\cdot\mathrm{p}\}, \{I_x\}$ and $\{I_y\}$ where $\{\nabla I\}$ is a vector containing the gradient of the intensity at each pixel, and I_x and I_y are the gradients of the intensity in the reference image in the *x* and *y* directions.

10. (Original) The method of claim 2, wherein the step of recovering camera rotation vectors includes solving the following equation

$$P_\Omega\left(H^T S^{(1)} - \Psi w\right) = 0 \text{ for the 3-vector w.}$$

11. (Original) The method of claim 9 wherein, the step of computing a second projection matrix includes computing a $N_p \times N_p$ projection matrix P_T, which is block diagonal with zero entries between different smoothing windows and annihilates $\left(H^T S^{(1)}\right) - \Psi w$ where w is the vector recovered previously.

A4

12. (Original) The method of claim 2 wherein, the step of recovering the direction of camera translation includes solving for the direction of translation $\hat{T}$ via

$$P_{\hat{T}}\left(-\hat{T}_x\{I_x\} - -\hat{T}_y\{I_y\} + \hat{T}_z\{p \cdot \nabla I\}\right) = 0.$$

13. (Original) The method of claim 3 wherein, step (f) includes, recovering Z_n via

$$\left(H^T S^{(1)}\right)_n - [\Psi w]_n = Z_n^{-1}\left(\hat{T}_z p_n - \lfloor\hat{T}\rfloor_2\right) \cdot \nabla I_n \text{ where } \lfloor\hat{T}\rfloor_2 \text{ represents } \hat{T}_x \hat{T}_y \text{ the}$$

x and y component of the translation direction.

14. (New) A computer-readable medium that includes instructions for recovering scene structure and camera motion direction from successive image data obtained from a multi-image sequence containing a reference image and one or more successive images, whereby said one or more successive images of the sequence are taken at one or more

successive different perspectives by translating and/or rotating the camera wherein said instructions, when executed by a processor, cause the processor to:

a) compare a reference image to successive images, wherein the successive images is taken by translating or rotating the camera;

b) determine the image data shift for the successive images with respect to the reference image;

c) construct a shift data representation that incorporates the image data shifts for each image including a first data record corresponding to the 3D structure and a second data record corresponding to the camera motion;

e) divide the successive images into windows;

f) determine the direction of the camera motion and 3D structure from the first and second data record between the windows.

A4

15. (New) A computer system for recovering scene structure and camera motion direction from image data obtained from a multi-image sequence containing a reference image and one or more successive images, whereby said one or more successive images of the sequence are taken at one or more successive different perspectives by translating and/or rotating the camera, comprising:

a processor;

a video source whose output is digitized into a pixel map by a digitizer, said output is sent in electronic form via a system bus for access by main memory;

a display means;

a user interaction means for selecting items on the display means;

a storage device in communication with said processor, said storage device stores program code for programming said processor to perform a method comprising the steps of:

(a) accessing stored digitized video by the processor from main memory;

(b) determining the image data shift for the successive image with respect to the reference image;

(c) constructing a shift data representation that incorporates the image data shifts for each image including a first data record corresponding to the 3D structure and a second data record corresponding to the camera motion;

(d) dividing the successive images into windows;

(e) determining the direction of the camera motion and 3D structure from the first and second data records between the windows.

A4